

Des algorithmes pour minimiser l'espérance de temps pour trouver une place de stationnement en zone urbaine

Asma Houissa^{1,2}, Dominique Barth¹, Nadège Faul², Thierry Mautor¹

¹ Laboratoire DAVID, Université de Versailles-Saint Quentin, 45 Avenue des Etats Unis, 78000 Versailles, France

{asma.houissa, dominique.barth, thierry.mautor}@uvsq.fr

² Institut Védécom, 77 rue des chnatiers, 78000 Versailles, France

{asma.houissa, nadege.faul}@vedecom.fr

Mots-clés : *stationnement intelligent, recherche de chemins*

1 Introduction

Le stationnement urbain est un problème qui coûte en temps et en énergie. Une étude réalisée en 2005 [1] a évalué les coûts induits à la recherche d'une place de stationnement à 600 millions d'euros par an en France (en incluant le temps perdu, l'énergie, le surplus de trafic,...). C'est pourquoi le stationnement intelligent est un domaine de recherche qui croît très rapidement. L'émergence de nouveaux navigateurs et smartphones, ainsi que la proposition de nouvelles techniques de détection des lieux disponibles ont permis récemment à plusieurs recherches et applications d'être initiées, ce qui confirme l'innovation et l'originalité de notre travail. Parmi les applications mobiles, nous pouvons citer Apila [2] qui permet un échange de places de stationnement disponibles en temps réel entre utilisateurs. L'application mobile Parknav [3] propose une solution de stationnement pour les automobilistes qui veulent se garer rapidement en voirie. Une autre application mobile OPnGO [4] repère pour le conducteur les parkings autour de lui, compare les prix et trouve la meilleure place.

Dans une ville où aucune infrastructure de capteurs n'est déployée dans chaque lieu mais seulement un système de comptage à chaque intersection, nous montrons qu'il est possible de proposer une méthode efficace qui détermine un itinéraire qui minimise l'espérance de temps pour trouver une place de stationnement disponible. Pour cela, nous modélisons d'abord la zone urbaine par un graphe. Ensuite, nous mettons en œuvre des algorithmes pour minimiser l'espérance de temps de trouver une place de stationnement en zone urbaine. Nous évaluons la performance des algorithmes par des simulations réalistes extraites de la carte de Versailles.

2 Description du problème

Le modèle que nous proposons considère les segments de rue (c'est-à-dire la portion maximale de rues ne contenant aucune intersection). Ainsi, une zone urbaine est modélisée par un graphe orienté $G(S, A)$. L'ensemble des noeuds S représente les intersections de la rue. Chaque arc $a = (x, y)$ dans A désigne un segment de rue de l'intersection x à l'intersection y . Dans ce modèle, nous considérons différentes métriques caractérisant chaque nœud et chaque arc dans le graphe. En particulier, sur chaque arc a , nous avons le nombre de places de stationnement et la probabilité qu'il y ait au moins une place de stationnement libre.

Nous considérons maintenant un nœud DF dans G qui est la destination finale d'un conducteur qui cherche à se garer à proximité. Soit D_{Pmax} un paramètre indiquant la distance maximale que le conducteur accepte de marcher une fois sa voiture garée pour atteindre sa destination. Nous définissons l'orbite de stationnement $O(DF, D_{Pmax})$ sous la forme d'un sous-graphe de

G induit par tous les nœuds qui sont à une distance à pied inférieure ou égale à D_{Pmax} de la destination DF. Le paramètre T_{max} indique le temps maximum que le conducteur accepterait de passer à la recherche d'une place de stationnement dans la zone. Nous considérons également qu'un nœud PE dans cette orbite est le premier nœud à partir duquel le conducteur entre dans l'orbite. Etant donné une orbite O , une position de départ PE, une destination finale DF et deux paramètres T_{max} et D_{Pmax} , nous définissons un itinéraire par un chemin dans l'orbite $O(DF, D_{Pmax})$ dont le sommet de départ est PE et la durée maximale dépasse tout juste T_{max} . *Problème* : Déterminer dans l'orbite $O(DF, D_{Pmax})$ un itinéraire à partir de PE qui minimise l'espérance de temps pour stationner

3 Méthodes de résolution développées

Nous proposons tout d'abord un algorithme glouton pour trouver un itinéraire qui minimise l'espérance de temps de trouver une place de stationnement. Cet algorithme est paramétré par une valeur de profondeur de recherche notée n . A partir d'un nœud de départ, nous énumérons, l'ensemble des itinéraires partiels par une évaluation arborescente complète dont on limite la profondeur par un paramètre n , afin d'identifier le chemin de n portions de rues qui minimise l'espérance. Le premier arc de ce chemin est choisi, puis on réitère le processus jusqu'à ce que la durée T_{max} soit atteinte. L'évaluation des performances de cet algorithme montre qu'il est extrêmement difficile de trouver une place de stationnement dans la zone considérée de Versailles. La solution optimale donnée par $n = 30$ (énumération complète de tous les itinéraires) montre que même en suivant le meilleur itinéraire, il faut en moyenne plus d'un quart d'heure pour stationner. Le second point est que les solutions données avec $n = 5$, $n = 4$ ou même $n = 3$, ne sont pas vraiment loin de cette solution optimale. Ces solutions sont rapidement calculées (en particulier $n = 3$) et fournissent des itinéraires où le temps moyen d'attente pour se garer n'est pas très élevé par rapport au meilleur itinéraire. Dans cette partie, il est possible aussi de modéliser la recherche de stationnement par le processus décisionnel de Markov(MDP)[5] et de résoudre cette problématique par la programmation dynamique. Toutefois, nous nous sommes intéressés aux algorithmes d'apprentissage par renforcement. Nous avons proposé une méthode d'apprentissage par renforcement basée sur la méthode LRI [6] (Linear Reward Inaction). Dans ce modèle, chaque agent modélisant une intersection apprend la meilleure portion de rue suivante. A chaque étape, l'ensemble des décisions prises par les agents génère un itinéraire dont l'espérance est la base de la mise à jour des paramètres d'apprentissage. Nous répétons cette étape jusqu'à atteindre une relative stabilité des vecteurs d'apprentissage des sommet de plus faible profondeur arborescente, afin de choisir (comme précédemment) la portion suivante. Les temps d'exécution et les performances de cette approche seront comparés à ceux de la méthode gloutonne.

Références

- [1] E.Grantelet and A.Lefauconnier. La recherche d'une place de stationnement : stratégies, nuisances associées, enjeux pour la gestion du stationnement en france, company report, Sareco,2005.
- [2] Apila :Application mobile, www.apila.fr, company website, November 2016.
- [3] Parknav : Application mobile,www.parknav.com, company website, November 2016
- [4] OPnGO : Application mobile,www.opngo.com, company website, November 2016
- [5] S.Tang, T.Rambha, R.Hatridge, S.Boyles, A.Unnikrishnan.Modeling Parking Search on a Network by Using Stochastic Shortest Paths with History Dependence, Journal of the Transportation Research Board, Volume 2467, pp. 73-79, January 2014.
- [6] K.S. Narendra and M.A.L. Thathachar. Learning Automata : An introduction. Prentice Hall, 1989